

⑫ 公開特許公報(A)

平1-124060

⑮ Int.Cl.⁴

識別記号

庁内整理番号

⑬ 公開 平成1年(1989)5月16日

G 06 F 15/60

3 6 0

A-6615-5B

審査請求 未請求 発明の数 1 (全7頁)

⑭ 発明の名称 ネットワーク図作成装置

⑯ 特 願 昭62-282846

⑰ 出 願 昭62(1987)11月9日

⑱ 発 明 者 中 野 利 彦 茨城県日立市大みか町5丁目2番1号 株式会社日立製作所大みか工場内

⑲ 発 明 者 森 清 三 茨城県日立市大みか町5丁目2番1号 株式会社日立製作所大みか工場内

⑳ 出 願 人 株式会社日立製作所 東京都千代田区神田駿河台4丁目6番地

㉑ 代 理 人 弁理士 鶴沼 辰之 外1名

明 細 書

1. 発明の名称

ネットワーク図作成装置

2. 特許請求の範囲

1. ネットワーク状の関連を持つデータを入力する装置と、前記データの関連をネットワーク図としてレイアウトするレイアウト装置と、レイアウトされたネットワーク図を出力する出力装置とからなるネットワーク図作成装置において、

前記レイアウト装置が、

前記データのネットワーク状の関連を有向グラフに変換する手段と、

前記有向グラフに閉ループが存在するときに当該閉ループの逆向き有向グラフを削除する手段と、

閉ループが除かれた有向グラフの最長パスを調べる手段と、

前記最長パスの下位ノードから関連を持つノードを前記最長パスに順次付け加え前記有向グラフを木構造に変換する手段と、

(1)

前記木構造の各ノードをレイアウトし各ノード間を線で結合する手段と

を備えたことを特徴とするネットワーク図作成装置。

2. 特許請求の範囲第1項において、

前記有向グラフに変換する手段が、各有効グラフの意味のつながりの強さを記憶する手段を含み、

前記有向グラフの最長パスを調べる手段が、分岐ノードで前記意味のつながりの強い有向グラフを優先的に採用する手段を含むことを特徴とするネットワーク図作成装置。

3. 発明の詳細な説明

〔産業上の利用分野〕

本発明は、ネットワーク図作成装置に係り、特に、見易さを確保しながら作業効率を大幅に高めたネットワーク図自動レイアウト作成装置に関するものである。

〔従来の技術〕

ネットワーク図作成作業においては、従来、次

(2)

の3つの方式が用いられてきた。

第1の方式は、レイアウトに必要なすべての情報をオペレータ（作業員）に問合せ、入力されたレイアウトの情報に基づきレイアウトを実行するCADシステムのような手法である。

第2の方式は、線以外の部分をオペレータに問合せ、入力されたレイアウト情報に基づき、それらのレイアウトを実行し、レイアウトが終了した時点で線により結合する手法である。第2図と第3図とを参照して、第2の方式を説明する。第2図において、A～Jはレイアウトすべきノード、a～oはノード間を結合する線である。四角で囲んだ数字はノード間の結び付きの強さを表す。この数字が大きいくほど結び付きが強い。具体的な操作手順を第3図に示す。A段階で、まず、レイアウトを行うシステムを起動し、レイアウトするスペースを表示する。B段階で、表示したスペース上に、ノードA～I等を設定していく。この時、システムは各ノードとそれを表示すべき位置とを併せて記憶する。C段階で、システムは各ノード

(3)

画面において、対話形式でデータを変更しレイアウト操作を実行しようとしても、応答が遅く使用に耐えない。また、各ノードのつながりの強さ等を考慮しないで、機械的にレイアウトすることから、本来は意味的にまとめてレイアウトしたいノードが離れてレイアウトされることもあり、図の見易さが損なわれてしまう欠点があつた。

本発明の目的は、ネットワークの意味的な情報のみを用いて、試行錯誤を実質上伴わず、意味的なつながりを持つノードを近くにレイアウト可能なネットワーク図作成装置を提供することである。
〔問題点を解決するための手段〕

上記目的は、ネットワーク状に定義されたノードの関係を木構造の関係として一旦とらえ、ノード全体をレイアウトし、それらノードを線で結合することにより達成される。

すなわち、本発明は、上記目的を達成するために、ネットワーク状の関連を持つデータを入力する装置と、それらデータの関連をネットワーク図としてレイアウトするレイアウト装置と、レイア

(5)

間に必要な線を自動的に生成する。

第3の方式は、レイアウト上のすべての組合せを調べ、例えば面積が最小となるレイアウトを実行するLSI-CADのような手法である。

〔発明が解決しようとする問題点〕

上記第1および第2の方式においては、ネットワークのノード間の関係の外に、ノードのレイアウト上の位置情報についても記憶する必要がある。このため、ノードの追加、削除が発生した場合、レイアウトと位置情報の記憶とを再度やり直す必要がある。また、レイアウトするスペースが変更になる場合や、表示面積または表示手段が異なる機器に同一の情報を出力するためそれぞれに対して最適なレイアウトを実行する必要が発生した場合は、それぞれレイアウトをやり直すだけでなく、それらレイアウトを機器の数だけすべて記憶しておかなければならない。

一方、第3の方式では、レイアウト上で多数回の試行錯誤がなされ、最終レイアウトを決定するまでに、多くの時間を要する。したがって、端末

(4)

ウトされたネットワーク図を出力する出力装置とで構成されたネットワーク図作成装置において、レイアウト装置が、データのネットワーク状の関連を有向グラフに変換する手段と、有向グラフに閉ループが存在するときにその閉ループの逆向き有向グラフを削除する手段と、閉ループが除かれた有向グラフの最長パスを調べる手段と、最長パスの下位ノードから関連を持つノードをその最長パスに順次付け加え有向グラフを木構造に変換する手段と、木構造の各ノードをレイアウトし各ノード間を線で結合する手段とを備えたネットワーク図作成装置を提案するものである。

有向グラフに変換する手段には、各有効グラフの意味のつながりの強さを記憶する手段を備え、有向グラフの最長パスを調べる手段には、分岐ノードで前記意味のつながりの強い有向グラフを優先的に採用する手段を備えることが望ましい。

〔作用〕

ネットワークから木構造に変換するときに、ネットワーク状の仕様を閉ループに含まない有向グ

(6)

ラフとしてとらえ、次に、有向グラフの最長パスを木の幹とする。この幹上に在るノードの最下位ノードと関係のあるノードから順次この幹にノードを付与していく。本方式により、ネットワークの線の矢印の方向が一定方向となる。矢印が一定方向化すると、オペレータがネットワーク図を見るときの見方とレイアウトの流れの方向とを一致させることができる。また、木構造の最長パス決定時に、近くにレイアウトする必要があるノードについては、ノード間の線に優先度を設定し、優先度の高いノードを優先的に幹に採用する。

このように、ネットワークのレイアウトを木構造のレイアウトに帰着させたため、見易さを確保したまま、高速レイアウトが可能となる。すなわち、従来の方法では、ノード数が増加するにつれて、レイアウトするための計算量が指数関数的に激増するが、本発明では、ノード数に比例した計算量で済むことになる。

また、最長パスに基づき木構造化するので、ノード間の線を自然な形にできる。

(7)

また、 $A \rightarrow B \rightarrow D \rightarrow G$ とした場合は、C, E, J, Hがそれぞれ該当ノードとなる。それを第6図(b)に示す。

さらに、ノードがすべて幹とつながるまで、幹に付けるノードを調べ続ける。

〔実施例〕

次に、本発明によるネットワーク図作成装置の一実施例の構成を第1図により説明する。

本装置は、入力装置10とレイアウト装置12と出力装置28とからなる。レイアウト装置12は、入力装置ドライバ14と演算部16と出力装置ドライバ18とワーキングメモリ26とを備えている。演算部16は仕様入力部20とレイアウト部22とを含み、ワーキングメモリ26は仕様記憶テーブル24を含んでいる。

入力装置ドライバ14は、入力装置10からのデータを演算部16が理解可能なデータ形式に変換する。一方、出力装置ドライバ18は、出力装置(CRT表示装置やXYプロッタ等)28の動作に好適な信号形式で、レイアウト図等を出力す

(9)

第2図の例について、第4図から第6図を参照し、本発明における有向グラフから木構造への変換の概要を説明する。

ここでは、各ノード間の意味的なつながり情報とともにノードが入力され、有向グラフとして仮の矢印付けが終了し、第2図が得られているものとする。

まず、第2図で閉ループとなつている $D \rightarrow g \rightarrow E \rightarrow h \rightarrow D$ の線hをカットし、閉ループの無い有向グラフとする。カット後の有向グラフを第4図に示す。

次に、第4図の有向グラフにおいて、最長パスを調べる。最長パスは、 $A \rightarrow B \rightarrow D \rightarrow E \rightarrow G$ である。ただし、Dで優先度(fが1でgが0)を評価した場合、 $A \rightarrow B \rightarrow D \rightarrow G$ となる。これを木構造の幹とする。木構造の幹を第5図に示す。

そこで、木構造の葉からルートの順に、幹に付けるノードを調べる。第5図で、 $A \rightarrow B \rightarrow D \rightarrow E \rightarrow G$ を幹にした場合は、C, F, J, Hがそれぞれ該当ノードとなる。それを第6図(a)に示す。

(8)

る。

なお、入力装置10側の機器と出力装置28側の機器とは、一つのCRT表示装置やキーボード等を共用しても良く、入力データと出力図形とをマルチウインド形式で表示することも自由である。

入力装置10から入力されたレイアウトすべき対象に関するデータは、入力装置ドライバ14と仕様入力部20とを介して、仕様記憶テーブル24に格納される。レイアウト部22はそのデータを用いて、レイアウト図を演算し、出力装置ドライバ18を介して、結果を出力装置28に出力する。

仕様記憶テーブル24の一例を第7図に示す。この仕様記憶テーブル24は、いわゆる画像メモリではなく、表形式である。仕様記憶テーブル24は、作成すべきネットワークのノードを縦軸と横軸のそれぞれに記憶し、これらノード間にレイアウト線が在ることを表の交点にフラグ1を立てて表す。すなわち、縦軸の始点ノード301(例えばJ点)から終点ノード(例えばE点)の

(10)

方向に線が存在することを交点303に1を立てて表し、無い場合は「-」を記憶する。また、線が存在する場合は、線の優先度304を整数1, 0等で記憶する。この数が大きいほど優先度が高い。例えば、D→Gが1で、D→Eが0であるから、D→Gの優先度が高い。

レイアウト部22は、第7図の仕様記憶テーブル24に基づき、第8図のレイアウトテーブルを用い、第9図のアルゴリズムにより動作する。

第8図は、第2図や第4図等のレイアウトを表形式で表現したレイアウトテーブルである。レイアウトテーブルは、ネットワークを構成するノードを縦軸及び横軸に持つ。縦軸と横軸の交点には、各線の先にいくつのノードが存在するかを記憶するノード数エリア401と、木構造の幹として用いた線を区別する区分フラグ402とが記憶される。例えば、ノードAからノードB方向を見ると、B, D, E, Fの4個のノードが在るから、始点ノードAと終点ノードBとの交点にノード数エリアに記憶する数は4となる。同様にAからJ方向

(11)

を終点とする線が記憶される。

ステップ120では、第8図のレイアウトテーブルにおいて、ステップ110で記憶した線と対応する部分のノード数エリアの値を1とする。例えば、B→Cの交点に1が記憶される。

ステップ130では、第8図レイアウトテーブルのノード数エリア401の値がすべて決定されたかどうかを判断する。

まだの場合は、ステップ140に行き、レイアウトテーブルにおいて、線のノード数エリアの値がすべて決定された始点ノードを記憶する。

ステップ150では、ステップ140で記憶した各始点ノードから出る線の内ノード数エリアの最大値を記憶する。

ステップ160では、ステップ140で記憶した始点ノードを終点とする線のノード数エリアの値をステップ150で記憶した値+1とする。すなわち、より上位のノードであることを記憶する。

このような手順を順次続けると、ステップ130でレイアウトテーブルのノード数エリアの値がす

(13)

では、J, E, Fの3となる。ただし、EからDの逆向き有向グラフで削除された部分には、0を記憶する。区分フラグ402を、1の始点ノードAから、2のB, 3のD, 4のEとたどつていけば、幹となつたノードが分かる。

次に、第9図のアルゴリズムを説明する。入力装置からネットワークを構成するノードの相互関連を示す情報を入力した第7図の状態（関連するノードの交点にフラグ1を立て、優先度の数値も入力した状態）を前提とする。

ステップ100では、ネットワークを検索し、閉ループとなつている線に対するレイアウトテーブルのノード数エリアの値を0とする。これは、有向グラフに閉ループが存在するときに、その閉ループの逆向き有向グラフを削除する段階である。

ステップ110では、第7図に示した仕様記憶テーブルにおいて線が出ていない始点ノードを見つけ、そのノードを終点ノードとしている線をすべて記憶する。線が出ていない始点ノードは、C, F, G, Iである。したがって、これらのノード

(12)

べて決定された状態に至る。

ステップ110からステップ160までが、入力データのネットワーク状の関連を有向グラフに変換する段階である。

ステップ170では、線を持たない終点ノードのノード名を記憶する。この場合は、Aである。

ステップ180では、ステップ170のノードを始点ノードとする線の内優先度が高くノード数エリアの値が最大のものの区分フラグを1とする。A→Bが該当する。

ステップ190では、ステップ180で設定したノード名を記憶し、ノード数エリアの値がすべて定義されていないノードまでステップ180を繰り返し行う（区分フラグはカウントアップしていく）。

ステップ170からステップ190までは、木構造の幹を作成する段階である。ここでは、A→B→D→Eが幹として作成されている。

ステップ200では、終点ノードの内一つも区分フラグが設定されていないものを探す。すな

(14)

わち、最下位ノードの方から木構造化する。

ステップ210では、ステップ200で見つけた終点ノードの線に対応する始点ノードをすべて調べる。始点ノードの区分フラグが定義されているものが一つの場合は、その始点ノードを記憶する。始点ノードの区分フラグが定義されているものが二つ以上の場合は、区分フラグの数値が小さいものの始点ノードを記憶する。木構造では、一つのノードから複数の線は出ていけるが、複数の線は入れないからである。

ステップ220では、ステップ210で記憶した始点ノードに対応する線のステップ200で探した終点ノードの区分フラグの値をステップ210の区分フラグの値+1に設定する。すなわち、階層が下がるにつれて区分フラグの値を増やしていき、有向グラフを木構造に変換する。

ステップ200からステップ220までは、最長パスに下位ノードから関連を持つノードを最長パスに順次付け加え有向グラフを木構造に変換する段階である。

(15)

す図、第4図は有向グラフの一例を示す図、第5図は木構造の幹を示す図、第6図は葉から幹に付けるノードを調べる状態を示す図、第7図は仕様記憶テーブルの一例を示す図、第8図はレイアウトテーブルを示す図、第9図は本発明レイアウト装置のアルゴリズムを示す図である。

10…入力装置、12…レイアウト装置、16…演算部、20…仕様入力部、22…レイアウト部、24…仕様記憶テーブル、26…ワーキングメモリ、28…出力装置、301…始点ノード、302…終点ノード、303…交点、304…優先度、401…ノード数エリア、402…区分フラグ。

代理人 弁理士 鶴沼辰之

ステップ230では、レイアウトテーブルに基づきノードをレイアウトする。

ステップ240では、レイアウトテーブルに基づき線を引く。

こうして、レイアウト図が完成する。

レイアウト図のデータは、出力装置に送られ、表示される。

〔発明の効果〕

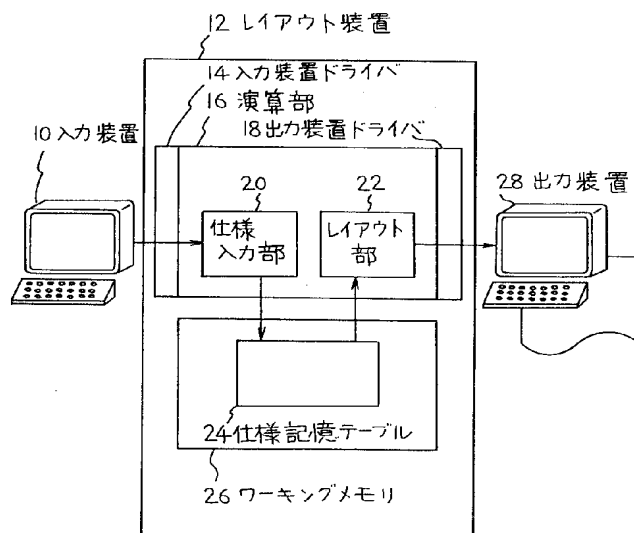
本発明によれば、見易さを確保したままネットワーク図を自動レイアウトすることが可能となるので、例えばソフトウェア仕様のように意味的な関係しか定義されていないものでも、ネットワーク図として出力でき、従来の画像情報として入力し管理し修正する方式と比較して、作業効率が大幅に改善される。

4. 図面の簡単な説明

第1図は本発明によるネットワーク図作成装置の一実施例の構成を示すブロック図、第2図はレイアウトすべき情報の一例を有向グラフ形式で示す図、第3図は従来のレイアウト手順の一例を示す図、

(16)

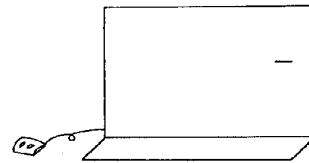
第1図



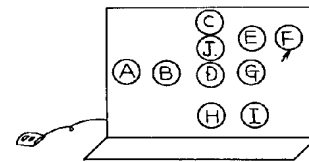
(17)

第 3 図

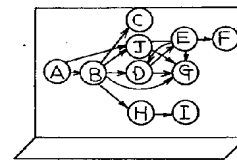
(A)



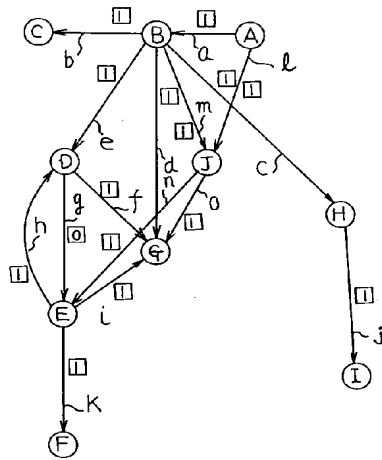
(B)



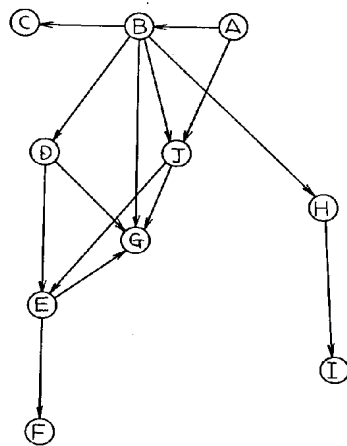
(C)



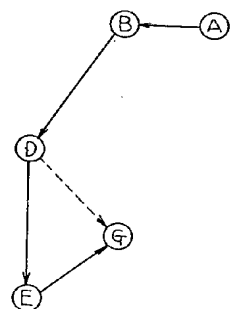
第 2 図



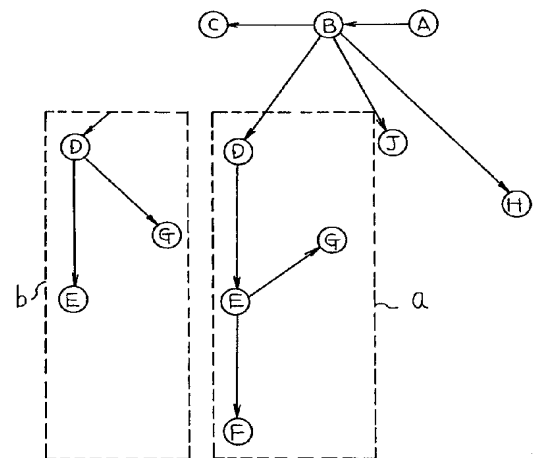
第 4 図



第 5 図



第 6 図



第 7 図

301 始点ノード	302 終点ノード	A	B	C	D	E	F	G	H	I	J
A		-	1 ₍₀₎	-	-	-	-	-	-	-	1 ₍₀₎
B		-	-	1 ₍₀₎	1 ₍₀₎	-	-	1 ₍₀₎	1 ₍₀₎	-	1 ₍₀₎
C		-	-	-	-	-	-	-	-	-	-
D		-	-	-	-	1 ₍₀₎	-	1 ₍₀₎	-	-	-
E		-	-	-	1 ₍₀₎	-	1 ₍₀₎	1 ₍₀₎	-	-	-
F		-	-	-	-	-	-	-	-	-	-
G		-	-	-	-	-	-	-	-	-	-
H		-	-	-	-	-	-	-	-	1 ₍₀₎	-
I		-	-	-	-	-	-	-	-	-	-
J		-	-	-	-	1 ₍₀₎	-	1 ₍₀₎	-	-	-

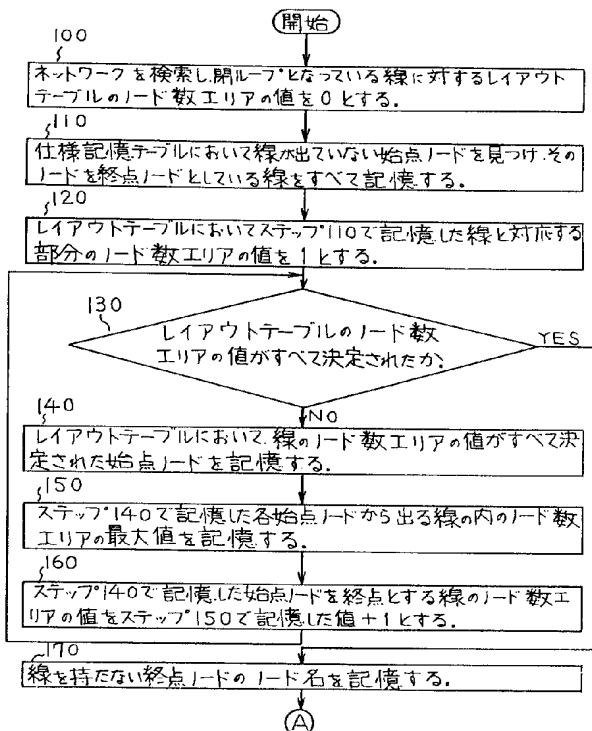
301…始点ノード
302…終点ノード
303…交点
304…優先度

第 8 図

401 始点ノード	402 終点ノード	A	B	C	D	E	F	G	H	I	J
A		-	4	-	-	-	-	-	-	-	3
B		-	-	-	2	-	-	-	-	-	3
C		-	-	1	3	-	-	1	2	-	3
D		-	-	-	-	3	-	-	-	-	-
E		-	-	-	-	2	-	1	4	-	-
F		-	-	-	0	-	1	1	-	-	-
G		-	-	-	-	-	-	-	-	-	-
H		-	-	-	-	-	-	-	-	-	-
I		-	-	-	-	-	-	-	-	1	-
J		-	-	-	-	2	-	1	-	-	-

401…ノード数エリア
402…区分フラグ

第 9 図(その 1)



第 9 図(その 2)

